AF
ZJW

PATENT APPLICATION

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Applicants: Kim et al.

Examiner: Patel, G.

Serial No.: 09/338,473

Group: Art Unit 2655

Filed: June 22, 1999

Docket: 8836-116 (IB8187-US)

For: **APPARATUS FOR CONTROLLING MULTI-WORD STACK OPERATIONS IN DIGITAL DATA PROCESSORS**

To: Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450
Mail Stop-Appeal Brief Patents

Request for Reinstatement of Appeal

This paper is respectfully submitted in support of Applicants' request for reinstatment of the Appeal that was commenced by Notice of Appeal filed June 14, 2004. The Appeal in this application was dismissed by the Examiner on the ground that the Appeal Brief filed on 8/29/05 allegedly failed to comply with the requirements of 37 CFR 41.37. Upon dismissal of the Appeal, the Examiner proceeded with an Examiner's Amendment to cancel claims 1-8 (which were being appealed) and issue a Notice of Allowance (mailed on 10/28/05) with regard to allowed claims 9-25. In support of Applicants' request for reinstatment of the Appeal, Applicants hereby submit a new Appeal Brief and respectfully request that the Appeal be reinstated and proceed with the new Appeal Brief submitted herewith, for the following reasons.

CERTIFICATE OF MAILING 37 C.F.R. § 1.8(a)

I hereby certify that this correspondence and the other documents that are indicated as being submitted herewith were deposited with the United States Postal Service as first class mail, postpaid in an envelope, addressed to the Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, Mail Stop Appeal Brief Patents on the date indicated below

Date: 1/23/06

Frank DeRosa

Since the time that the Appeal was dismissed, the undersigned has had several interviews with the Examiner, Gautam Patel, and the Examiner's SPE, Hoa T. Nguyen, seeking clarification as to the basis for the dismissal, as the original grounds set forth in the Notice of the dismissal of the Appeal were vague and unclear. From these interviews, the undersigned has come to learn that the Appeal was dismissed for the following reasons:

(i) The Appeal Brief filed on 8/29/05 included an extra "Introduction" section, which is not specifically listed in 37 CFR 41.37 (c) as a required section; and

(ii) The Appeal Brief filed on 8/29/05 did not contain a concise explanation of the subject matter defined in independent claims 2 and 6 involved in the appeal with reference to specification by page and line number (although each element of claims 2 and 6 were indeed described with specific reference to FIG. 2 and reference characters of elements depicted in FIG. 2).

The Applicants respectfully submit that the above listed reasons (i) and (ii) are, at most, merely informal technicalities of non-compliance that should not have led to the dismissal of the Appeal in this application. To begin, with regard to item (i), the inclusion of the Introduction section, which sets forth the basis for the appeal (e.g., dates of the Final Office Action and Notice of Appeal), is merely informative, and there is nothing in the CFR Rules that prohibits the inclusion of such information.

Further, with regard to item (ii), although the summary description of claims 2 and 6 did not technically include a recitation to page and line numbers of the specification, each element of claims 2 and 6 was described with specific reference to reference characters of elements depicted in FIG. 2. As noted in section 1205.2 of the MPEP, the "Summary of claimed subject matter" section is to enable the Board to more quickly determine where the claimed subject matter is described in the specification. In this regard, Applicants' reference to specific reference

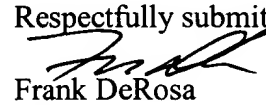
characters in FIG. 2 to describe the elements of claims 2 and 6 undoubtedly enables the Board to determine where the claimed subject matter is described in the specification, especially given that the Summary indicated that claims 1, 2 and 6 were being commonly described with reference to FIG. 2, and given that the corresponding page and line numbers in the specification for FIG. 2 were listed in the Summary section for claim 1.

In view of the above, Applicants' contend that the Appeal brief of 8/29/2005 was in substantial compliance with the content requirements specified in 37 CFR 41.37(C) and that Applicants should have been afforded an opportunity to remedy the minor technicalities in (i) and (ii), rather than have the Appeal dismissed.

In fact, the Examiner's SPE, Hoa T. Nguyen, acknowledged to the undersigned during an interview that the Appeal Brief was, at the very least, in substantial compliance with the content requirements of 37 CFR 41.37(C) and that reconsideration of the Appeal dismissal would be afforded to Applicants upon filing this Request for Reconsideration together with a new Appeal Brief addressing the minor non-compliances of (i) and (ii).

Accordingly, the new Appeal Brief submitted herewith has been revised to remove the "Introduction" section. Moreover, the "Summary of Claimed Subject Matter" section has been revised to include a specific reference to page and line numbers for the summary description of claims 2 and 6 (similar to that provided for claim 1). Therefore, in view of the above, Applicants respectfully request reinstatement of the Appeal in this application.

Respectfully submitted,



Frank DeRosa
Reg. No. 43,584

F. Chau & Associates, LLC
130 Woodbury Road
Woodbury, New York 11797
TEL.: (516) 692-8888
FAX: (516) 692-8889

PATENT APPLICATION

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Applicants: Kim et al.

Examiner: Patel, G.

Serial No.: 09/338,473

Group: Art Unit 2655

Filed: June 22, 1999

Docket: 8836-116 (IB8187-US)

For: **APPARATUS FOR CONTROLLING MULTI-WORD STACK OPERATIONS IN DIGITAL DATA PROCESSORS**

APPEAL BRIEF

Appeal from Group 2655

F. Chau & Associates, LLC
130 Woodbury Road
Woodbury, New York 11797
TEL: (516) 692-8888
FAX: (516) 692-8889
Attorneys for Appellant

TABLE OF CONTENTS **Page(s)**

I. REAL PARTY IN INTEREST

The real party in interest for the above-identified application is Samsung Electronics Co., LTD., the assignee of the entire right, title and interest in and to the subject application by virtue of an assignment of recorded in the U.S. Patent and Trademark Office at reel/frame 010057/0008.

II. RELATED APPEALS AND INTERFERENCES

There are no Appeals or Interferences known to Applicant, Applicant's representatives or the Assignee, which would directly affect or be indirectly affected by or have a bearing on the Board's decision in the pending Appeal.

III. STATUS OF CLAIMS

Claims 1-25 are pending. Claims 1-8 stand rejected and are under appeal. Claims 9-25 have been indicated by the Examiner as being directed to allowable subject mater and are not on appeal herein. All pending claims are set forth in the attached Appendix. Rejected claims 1, 2 and 6 are independent claims. Claims 3-4 and 7-8 depend directly or indirectly from claims 2 and 6, respectively.

IV. STATUS OF AMENDMENTS

An after final amendment was filed on February 17, 2004 in response to the FINAL ACTION, but an Advisory Action mailed on May 20, 2004 (Paper No. 16) indicated that for purposes of Appeal, the Amendment would not be entered.

V. SUMMARY OF CLAIMED SUBJECT MATTER

The claimed inventions are generally directed to hardware stack memory devices having multi-bank stack storage frameworks and control mechanisms for performing multi-word PUSH or multi-word POP operations, for example. In particular, in accordance with the claimed inventions, a stack storage is divided into two or more separate memory Banks. A separate Bank Pointer is provided for each memory Bank to point to a location (e.g., top) of the corresponding memory Bank. Control mechanisms are provided for the multi-bank framework to insert bank address data into a plurality of Bank Pointers and then insert (Push) or remove (Pop) a data word from each of the memory banks at a given time to perform a multi-word Push or multi-word Pop operation.

Independent claims 1, 2 and 6 broadly embody the above inventive concepts. For purposes of illustration, the inventions of claims 1, 2 and 6 are reproduced below and will be described with reference to the exemplary Figure 2, for example, and corresponding text of Appellants' Specification (Spec.), for example, but nothing herein shall be deemed as a limitation on the scope of the invention.

Claim 1 recites

1. *A hardware stack, comprising:*

a stack storage comprising a plurality of banks each comprising storage locations; and

a stack pointer circuit comprising a bank pointer for each bank, wherein each bank pointer points to a storage location of a corresponding bank, and wherein the stack pointer circuit is responsive to at least one control signal to insert bank address data in at least two bank pointers to perform a multi-word push or multi-word pop operation.

With respect to claim 1, FIG. 2 depicts an exemplary *hardware stack* (500). The hardware stack (500) comprises a *stack storage* (300) *comprising a plurality of banks* (Bank 1,

Bank 2) each comprising storage locations. A *stack pointer circuit* (200) *comprises a bank pointer* (230, 240) for each bank (Bank 1, Bank 2), wherein each *bank pointer* (230) and (240) points to a storage location of a corresponding one of the Banks (Bank 1, Bank 2). The *stack pointer circuit* (200) is responsive to at least *one control signal* (e.g., decoding signals from instruction decoder (100)) to insert bank address data in at least two bank pointers (230) and (240) to perform a multi-word push or multi-word pop operation. **(See, e.g., Spec., page 7, lines 22 through Page 8, line 12).**

Claim 2 recites:

2. *A digital data processor, comprising:*

an instruction decoder for decoding an instruction and generating a plurality of decoding signals;

a stack storage comprising a plurality of banks each comprising storage locations for storing stack items;

a plurality of stack pointers, the stack pointers comprising a main stack pointer for pointing to a top location of the stack storage and a bank pointer for each bank, wherein each bank pointer points to a storage location of a corresponding bank based on the content of the main stack pointer; and

a controller responsive to at least one of the decoding signals for inserting bank address data into at least two bank pointers to perform a multi-word push or multi-word pop operation.

With respect to Claim 2, FIG. 2 depicts and exemplary *digital data processor*, which comprises *an instruction decoder* (100) for decoding an instruction and generating a plurality of decoding signals, a *stack storage* (300) comprising a plurality of banks (Bank 1, Bank 2) each comprising storage locations for storing stack items, *a plurality of stack pointers* including *a main stack pointer* (210) for pointing to a top location of the *stack storage* (300) and a bank

pointer (230) and (240) for each bank. Each bank pointer (230) and (240) points to a storage location of a corresponding bank based on the content of *the main stack pointer* (210). The *digital data processor* further comprises a *controller* (220) responsive to at least one of the decoding signals (output from the instruction decoder (100)) for inserting bank address data into at least two bank pointers (e.g., 230 and 240) to perform a multi-word push or multi-word pop operation. **(See, e.g., Spec., page 7, lines 22 through Page 8, line 12).**

Claim 6 recites:

6. *A digital data processor, comprising:*

a stack storage including a plurality of locations, wherein each of the locations of said stack storage is assigned to one of a first bank and a second bank;

a main stack pointer for pointing to a top location of said stack storage;

a first bank stack pointer for pointing to a location assigned to said first bank;

a second bank stack pointer for pointing to a location assigned to said second bank;

an instruction decoder for decoding a stack-based instruction and generating a plurality of decoding signals, each of the plurality of decoding signals denoting one of a one-word push operation, a one-word pop operation, a two-word push operation and a two-word pop operation; and

a stack pointer control logic circuit for controlling said first and second bank stack pointers in response to at least one of the decoding signals to insert bank address data into the first and second bank stack pointers based on the content of the main stack pointer to perform a multi-word push or multi-word pop operation.

With respect to Claim 6, FIG. 2 depicts an exemplary *digital data processor*, which comprises a *stack storage* (300) including a plurality of locations, wherein each of the locations of said stack storage is assigned to one of *a first bank* (Bank 1) and *a second bank* (Bank 2), a

main stack pointer (210) for pointing to a top location of said *stack storage* (300) , *a first bank stack pointer* (240) for pointing to a location assigned to said first bank, a *second bank stack pointer* (230) for pointing to a location assigned to said second bank, an *instruction decoder* (100) for decoding a stack-based instruction and generating a plurality of decoding signals, each of the plurality of decoding signals denoting one of a one-word push operation, a one-word pop operation, a two-word push operation and a two-word pop operation, and a *stack pointer control logic circuit* (220) for controlling *said first and second bank stack pointers* (230, 240) in response to at least one of the decoding signals to insert bank address data into the first and second bank stack pointers based on the content of the *main stack pointer* (210) to perform a multi-word push or multi-word pop operation. **(See, e.g., Spec., page 7, lines 22 through Page 8, line 12).**

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

A. Claims 1-8 stand rejected under 35 U.S.C. §103(a) as being unpatentable over U.S. Patent No. 5,287,309 to Kai in view of U.S. Patent No. 6,167,488 to Koppala.

VII. ARGUMENTS

A. The Combination of Kai and Koppala is *Legally Deficient* to Support a *Prima Facie* Case Of Obviousness Against any of Claims 1-8

In rejecting claims under 35 U.S.C. §103, the Examiner bears the initial burden of presenting a prima facie case of obviousness. In re Rijckaert, 9 F. 3d 1531, 1532 (Fed. Cir. 1993). The burden of presenting a prima facie case of obviousness is only satisfied by showing some objective teaching in the prior art or that knowledge generally available to one of ordinary

skill in the art would lead that individual to combine the relevant teachings of the references. In re Fine, 837 F.2d 1071, 1074 (Fed. Cir. 1988). A prima facie case of obviousness is established when the teachings of the prior art itself would appear to have suggested the claimed subject matter to one of ordinary skill in the art. In re Bell, 991 F.2d 781, 782 (Fed. Cir. 1993). The suggestion to combine the references should come from the prior art, and the Examiner cannot use hindsight gleaned from the invention itself to pick and choose among related disclosures in the prior art to arrive at the claimed invention. In re Fine, 837 F.2d at 1075. If the Examiner fails to establish a prima facie case, the rejection is improper and must be overturned. In re Rijckaert, 9F.3d at 1532 (citing In re Fine, 837 F.2d at 1074).

In the case at bar, Appellants contend that at the very least, the FINAL ACTION does not present a legally sufficient basis for establishing a *prima facie* case of obviousness of claims 1, 2 and 6 based on the combination of Kai and Koppala. In formulating the obviousness rejections for claims 1, 2 and 6 (as set forth in the FINAL ACTION), the Examiner relies primarily on the teachings of Kai (particularly, FIG. 2 and relevant description) as modified by the teachings of Koppala. However, the Examiner's obviousness analysis is based on *impermissible hindsight reasoning* as the combined teachings of Kai and Koppala fail to disclose or fairly suggest the inventions of claims 1, 2 or 6, as a whole. In fact, as will be explained below, the Examiner's purported "motivation" or suggestion for combining the teachings of Kai and Koppala is belied by the express teachings of Kai, which actually teaches away from the Examiner's proposed combination. The impropriety of the obviousness rejections will be readily apparent based on an understanding of the teachings of Kai, as relied on by the Examiner, and, thus, a brief explanation of Kai will first be provided.

In formulating the obviousness rejections, the Examiner cites Kai as disclosing (in FIG. 2) a stack memory that is divided into two Banks. However, as will be readily apparent, Kai's use of a two-bank stack memory is very much different from the claimed invention. In particular, Kai discloses (in FIG. 2) a stack memory (5) that includes two Banks, an ODD bank (11) (which is allocated odd addresses for all addresses in the stack memory) and an EVEN Bank (10) (which is allocated even addresses for all addresses in the stack memory) (See, e.g., Kai Col. 4, lines 14-45).

Kai discloses a method for using the two-Bank stack as a means to provide high-speed stack operations by simultaneously performing, with respect to an input address, a *one-word* PUSH (write) operation in one Bank and a *one-word* POP (read) operation in the other Bank in ***advance*** of determining which access operation should be executed (see, e.g., Col. 1, lines 55-60; and Col. 3, lines 5-12).

In other words, the motivation behind using the two-Bank stack in the Kai process is that a push (write) and pop (read) operation can be simultaneously performed with respect to an input address, wherein one bank is used for the Push operation and the other bank is used for the Pop operation. The Kai framework enables an effective increase in the stack access time because the stack access operation (read and write) is commenced prior to deciding whether the Push (write) or Pop (read) operation is actually requested with respect to the input address (see, e.g., Abstract, Col 4, lines 10-20). Ultimately, however, only one of the single-word stack accesses (e.g., push or pop) is validated (see, e.g., Abstract, Col. 6, lines 20-32) and the stack pointer SP is adjusted accordingly.

Furthermore, Kai discloses that the data input to both Banks is the ***same*** (Col. 4, lines 27-43). Kai discloses a single stack pointer (6) that stores an input address (SP) over all of the stack

memory for indicating data to be processed by the POP operation, and the address for indicating data to be processed by the PUSH operation is (SP+1). When the address in the stack pointer (6) is an even address, the POP operation is performed in the EVEN Bank and the PUSH operation is performed in the ODD BANK . When the address in the stack pointer (6) is an odd address, the POP operation is performed in the ODD bank and the PUSH operation is performed in the EVEN Bank. (See Kai, Col. 4, line 46 – Col. 5, line 5).

(1) Claim 1 is Not Obvious in View of Kai and Koppala

Appellants respectfully contend that the combined teachings of Kai and Koppala do not disclose or fairly suggest, as a whole, the invention of Claim 1. In particular, by way of example, the combination of Kai and Koppala does not disclose or suggest a hardware stack having stack storage that is divided into multiple Banks with separate corresponding Bank Pointers, much less control mechanisms to insert bank address data into multiple Bank Pointers and generate a single control signal to either insert (PUSH) or remove (POP) a data word from the separate memory Banks to perform a multi-word Push or a multi-word Pop operation, as essentially claimed in claim 1.

The obviousness rejection of claim 1 as set forth in the FINAL ACTION is based primarily on the teachings of Kai (particularly, FIG. 2 and relevant description) as modified by the teachings of Koppala. The Examiner's obvious analysis begins on page 2 of the FINAL ACTION by citing FIG. 2 of Kai as disclosing a stack storage comprising a plurality of banks. Although Kai discloses a two-Bank stack storage, Kai does not disclose or suggest performing multi-word POP or PUSH Operations using a plurality of banks. In fact, the Examiner essentially acknowledges at various sections of the FINAL ACTION that Kai does not disclose performing multi-word PUSH or POP operations using a multi-bank framework. For example,

the Examiner acknowledges on Page 3 (2nd paragraph) of the FINAL ACTION that Kai does not disclose that the single stack pointer can be used to perform a multi-word operation. Moreover, the Examiner acknowledges on Page 8 of the FINAL ACTION that Kai does not disclose a PUSH-PUSH or POP-POP operation.

In an effort to cure the deficiencies of Kai, the Examiner relies on Col. 24, lines 29-32 of Koppala, which generally discloses that "*the most common stack manipulation for stack based computing system is to pop the top two data words off of the stack and to push a data word onto the stop of the stack*". (See page 3 of the FINAL ACTION). The Examiner points to Kai (Col. 2, lines 18-28), which generally discloses that many variations to the Kai system can be derived by simple mathematical transformation on designation of the addresses. The Examiner further notes (on pages 3 and 8 of the FINAL ACTION) essentially that Koppala discloses that "*control signal can be used for inserting a two word or multi-word item into said stack storage and removing a two-word item from adjacent locations at a given time*".

Based essentially on the above, the Examiner concludes (on bottom of Page 3 of the FINAL ACTION) essentially that *it would have been obvious to one of ordinary skill in the art to have provided, capability to remove two-word item from the stack storage, to the circuit of Kai as taught and suggested by Koppala, because it would have provided a mechanism to execute either a two Push or two POP operation on the stack, thus making the stack operation much faster.*

It is respectfully submitted, however, that the Examiner's grounds for obviousness is nothing more *than impermissible hindsight reasoning* based on the teachings of the Appellants' current specification. Indeed, the Examiner generally acknowledges that neither Kai nor Koppala disclose the use of a multi-bank stack structure for performing multi-word PUSH or POP

operations. But, irrespective of Koppala's general disclosure (in Col. 24, lines 29-32) of removing (popping) two words from the stack, the Examiner offers no legally sufficient explanation as to why one of ordinary skill in the art would be motivated to combine the teachings of Kai and Koppala to derive the claimed inventions in the manner suggested by the Examiner..

Appellants contend that the Examiner's reliance on Koppala as teaching performing a multi-word pop or push "*from adjacent locations at a given time*", to modify the teachings of Kai, is misplaced and is inconsistent with the express operation of the Kai system. Indeed, Koppala's teaching of *removing a two-word item from adjacent locations at a given time* actually teaches against the Examiner's proposed combination of Kai and Koppala. For example, as noted above, Kai discloses that the POP data is always stored in an address smaller by "1" than the address for storing the PUSH data (see, e.g., Abstract) In other words, with the two-bank (ODD-EVEN) framework of Kai, the PUSH and POP data are always stored at adjacent locations. This is required in the Kai system in order to implement the simultaneous single-word PUSH and POP operations at the top of the stack, wherein adjacent storage locations (difference of "1") store the data for simultaneous POP and PUSH operations.

In this regard, the Examiner's basis for motivation to modify the primary reference Kai with the teachings of Koppala to meet the claimed limitations, is legally improper as a matter of law. As noted above, the Examiner essentially contends that it would have been obvious to one of ordinary skill in the art to apply the teachings of Koppala to the circuit of Kai to execute a two POP operation (removing a two-word item from the stack) instead of one PUSH and one POP operation by simply transforming the logic as suggest by Kai.

However, it is axiomatic that if a proposed modification would render a prior art invention being modified unsatisfactory for its intended purpose, then there is no suggestion or motivation to make the proposed modification. See, MPEP 2143.01, citing *In re Gordon*, 733 F.2d 900 (Fed. Cir. 1964). Furthermore, if the proposed modification or combination of the prior art would change the principle of operation of the prior art invention being modified, then the teachings of the references are not sufficient to render the claims *prima facie* obvious. See, MPEP 2143.01, citing *In re Ratti*, 270 F.2d 810 (CCPA 1959).

Here, to the extent that Examiner's basis for obviousness is grounded on the stack storage system of Kai being modified by the teachings of Koppala such that Kai can be implemented to execute a two-word PUSH or two-word POP operation using the two-bank stack, instead of a simultaneous single-word PUSH and a single-word POP operation, it is respectfully submitted that such basis is legally improper. Indeed, as noted above, the impetus and basic principle of operation behind the two bank stack system of Kai is to provide a high speed stack operation by simultaneously executing a POP operation in *one bank* and a PUSH operation in the *other bank* so as to access the stack (RAM) before the data process mode is decided (see, e.g., Abstract).

Examiner's proposed modification of the Kai system to perform a two-word PUSH or two-word POP operation - i.e., using each bank to perform a single-word PUSH operation at the same time to obtain a two-word PUSH, or using each bank to perform a single-word POP operation at the same time to obtain a two-word POP - would fundamentally change the principle and purpose of the Kai system. Indeed, in such instance, the Kai system would only be able to perform either a multi-word Read operation or a multi-word Write operation. This proposed modification of Kai clearly renders Kai unsatisfactory for its intended purpose and significantly changes the principle of operation of the Kai system, which is to *simultaneously*

perform both a read operation in one bank and a write operation in the other bank *before* the actual data access mode is determined. As noted above, the two-bank Kai system requires that the POP data is always stored in an address smaller by "1" than the address for storing the PUSH data. Furthermore, in order to implement the Kai access protocol, Kai discloses that the data input to both Banks is the ***same*** (Col. 4, lines 27-43).

Based on this, there is simply no sound technical reasoning or legally sufficient motivation for combining Koppala's teaching (as contended by Examiner) of performing a multi-word pop or push "*from adjacent locations at a given time*" to modify the Kai system. It appears that the Examiner has simply resorted to hindsight reasoning based on the Appellants' specification to reconstruct the claimed inventions based on the general disclosures of a two-bank stack by Kai and a two-word POP by Kopalla, while ignoring the inconsistency and incompatibility of modifying Kai with Kopalla which renders Kai with Kopalla un-combinable as a matter of law.

Accordingly, Claims 1 and non-obvious over the combination of Kai and Koppala.

(2) Claim 2 is Not Obvious in View of Kai and Koppala

The rejection of claim 2 is set forth on Page 4 of the FINAL ACTION. The Examiner's basis for rejecting Claim 2 appears to be the same for that of claim 1 and thus the above discussion for claim 1 is applied equally to Claim 2.

(3) Claim 6 is Not Obvious in View of Kai and Koppala

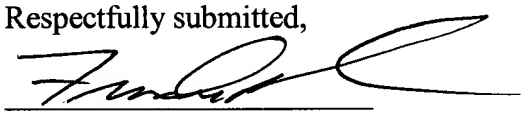
The rejection of claim 6 is set forth on Page 5 of the FINAL ACTION. The Examiner's basis for rejecting Claim 6 appears to be the same for that of claim 1 and thus the above discussion for claim 1 is applied equally to Claim 6. Furthermore, with regard to the rejection of claim 6, it appears that Examiner's reliance on Kai as disclosing a main stack pointer and first

and second bank stack pointers, is technically erroneous. In rejecting claim 6 (as well as claims 1 and 2), the Examiner relies on FIG. 2 of Kai as disclosing the claimed main stack pointer and first and second bank stack pointers. However, in contrast, Kai merely discloses in FIG. 2 a single stack pointer (6) which holds an address over all of the stack memory for indicating data to be processed by the POP operation (see, Col. 4, lines 58-63). The need for a single stack pointer (6) is due to the nature of the Kai system of, e.g., always simultaneously performing a concurrent Push and POP in adjacent locations at the top of the stack, thus obviating the need for separate main stack and multiple bank pointers.

B. CONCLUSION

Accordingly, for at least the above reasons, it is submitted that claims 1, 2 and 6 are patentable and non-obvious over the combination of Kai and Koppala. Accordingly, it is respectfully requested that the Board reverse all claim rejections under 35 U.S.C. § 103(a).

Respectfully submitted,



Frank DeRosa
Reg. No. 43,584

F. Chau & Associates, LLC
130 Woodbury Road
Woodbury, New York 11797
TEL: (516) 692-8888
FAX: (516) 692-8889

Claims Appendix

1. A hardware stack, comprising:

a stack storage comprising a plurality of banks each comprising storage locations; and

a stack pointer circuit comprising a bank pointer for each bank, wherein each bank pointer points to a storage location of a corresponding bank, and wherein the stack pointer circuit is responsive to at least one control signal to insert bank address data in at least two bank pointers to perform a multi-word push or multi-word pop operation.

2. A digital data processor, comprising:

an instruction decoder for decoding an instruction and generating a plurality of decoding signals;

a stack storage comprising a plurality of banks each comprising storage locations for storing stack items;

a plurality of stack pointers, the stack pointers comprising a main stack pointer for pointing to a top location of the stack storage and a bank pointer for each bank, wherein each bank pointer points to a storage location of a corresponding bank based on the content of the main stack pointer; and

a controller responsive to at least one of the decoding signals for inserting bank address data into at least two bank pointers to perform a multi-word push or multi-word pop operation.

3. The digital data processor of claim 2, wherein each location is configured for storing a one-word item.

4. The digital data processor of claim 3, wherein a two-word item is one of inserted into and removed from two adjacent locations at a time.

5. The digital data processor of claim 4, wherein said controller either increases or decreases the content of said main stack pointer by one when the decoding signals indicate a one-word stack operation; and wherein said controller either increases or decreases the content of said main stack pointer by two when the decoding signals indicate a two-word stack operation.

6. A digital data processor, comprising:

a stack storage including a plurality of locations, wherein each of the locations of said stack storage is assigned to one of a first bank and a second bank;

a main stack pointer for pointing to a top location of said stack storage;

a first bank stack pointer for pointing to a location assigned to said first bank;

a second bank stack pointer for pointing to a location assigned to said second bank;

an instruction decoder for decoding a stack-based instruction and generating a plurality of decoding signals, each of the plurality of decoding signals denoting one of a one-word push operation, a one-word pop operation, a two-word push operation and a two-word pop operation; and

a stack pointer control logic circuit for controlling said first and second bank stack pointers in response to at least one of the decoding signals to insert bank address data into the first and second bank stack pointers based on the content of the main stack pointer to perform a multi-word push or multi-word pop operation.

7. The digital data processor of claim 6, wherein said stack storage comprises 2^{n+1} locations, n being a positive integer, and wherein the first bank and the second bank each include 2^n locations.

8. The digital data processor of claim 6, wherein one of the first and second banks includes locations with addresses having a least significant bit of logic '0' and the other of the first and second banks includes locations with addresses having a least significant bit of logic '1'.

9. A digital data processor, comprising:

a stack storage including a plurality of locations, wherein each of the locations of said stack storage are assigned to one of a first bank and a second bank;

a main stack pointer for pointing to a top location of said stack storage;

a first bank stack pointer for pointing to a location assigned to said first bank;

a second bank stack pointer for pointing to a location assigned to said second bank;

an instruction decoder for decoding a stack-based instruction and generating a plurality of decoding signals, each of the plurality of decoding signals denoting one of a one-word push operation, a one-word pop operation, a two-word push operation and a two-word pop operation; and

a stack pointer control logic circuit for controlling said first and second bank stack pointers in response to at least one of the decoding signals to insert bank address data into the first and second bank stack pointers based on the content of the main stack pointer to perform a multi-word push or multi-word pop operation, wherein said stack pointer control logic circuit includes:

an adder for adding one of plurality of predetermined integers to a content of said main stack pointer in response to a first decoding signal from said instruction decoder;

a first selector for selecting for output one of the content of the main stack pointer and a content of said adder in response to a second decoding signal from said instruction decoder, wherein the output of said first selector comprises a high-order bit portion and a low-order bit portion;

a first control logic for generating a first control signal in response to the low-order bit portion of the output from said first selector and a third decoding signal from said instruction decoder;

a second control logic for generating a second control signal in response to the low-order bit portion of the output from said first selector and a fourth decoding signal from said instruction decoder;

an increment logic for incrementing the high-order bit portion of the output from said first selector;

a second selector for selecting one of the high-order bit portion of the output from said first selector and the output of said increment logic in response to the first control signal; and

a third selector for selecting one of the high-order bit portion of the output from said first selector and the output of said increment logic in response to the second control signal;

wherein the outputs of said second and third selectors are provided to said second and first bank stack pointers, respectively.

10. The digital data processor of claim 9, wherein said plurality of predetermined integers include one of +1, +2, -1 and -2; and wherein said increment logic increments the high-order bit portion of the output from said first selector by one.

11. The digital data processor of claim 9, wherein said main stack pointer is updated by the content of said adder.

12. The digital data processor of claim 9, wherein said low-order bit portion of the output from said first selector comprises the least significant bit of the output from the first selector.

13. The digital data processor of claim 12, wherein said first and second control logics alternately enable said second and third selectors, respectively, based on logic states of the least significant bit of the output from the first selector, during the one-word push and pop operations.

14. The digital data processor of claim 13, wherein said second selector is enabled when the least significant bit of the output from said first selector is logic >1'.

15. The digital data processor of claim 13, wherein said third selector is enabled when the least significant bit of the output from said first selector is logic >0'.

16. The digital data processor of claim 12, wherein said first and second control logics enable both of said second and third selectors, irrespective of logic states of the least significant bit of the output from said first selector, during the two-word push and pop operations.

17. A digital data processor, comprising:

a stack storage including a plurality of locations, wherein each of the locations of said stack storage are assigned to one of a first bank and a second bank;

a main stack pointer for pointing to a top location of said stack storage;

a first bank stack pointer for pointing to a location assigned to said first bank;

a second bank stack pointer for pointing to a location assigned to said second bank;

an instruction decoder for decoding a stack-based instruction and generating a plurality of decoding signals, each of the plurality of decoding signals denoting one of a one-word push operation, a one-word pop operation, a two-word push operation and a two-word pop operation; and

a stack pointer control logic circuit for controlling said first and second bank stack pointers in response to at least one of the decoding signals to insert bank address data into the first and second bank stack pointers based on the content of the main stack pointer to perform a multi-word push or multi-word pop operation, wherein said stack pointer control logic circuit comprises:

an adder adds one of a plurality of predetermined integers to a high-order bit portion of a content of said main stack pointer in response to a first decoding signal from said instruction decoder;

a control logic for generating one of a first, second, third, and fourth control signals, and combination thereof, in response to a low-order bit portion of the content of said main stack pointer and a second decoding signal from said instruction decoder;

a first selector for selecting one of the high-order bit portion of the content of said main stack pointer and an output of said adder in response to the first control signal;

a second selector for selecting one of the high-order bit portion of the content of said main stack pointer and the output of said adder in response to the second control signal; and

a third selector for selecting one of the high-order bit portion of the content of said main stack pointer and the output of said adder in response to the third control signal;

wherein outputs of said second and third selector are provided to said second and first bank stack pointers, respectively, and the low-order bit portion of the content of said main stack pointer is controlled by the fourth control signal.

18. The digital data processor of claim 17, wherein the plurality of predetermined integers comprise +1 and -1.

19. The digital data processor of claim 17, wherein said high-order bit portion of the content of said main stack pointer is updated by the content of said adder.

20. The digital data processor of claim 17, wherein said low-order bit portion of the content of said main stack pointer comprises a least significant bit of the content of said main stack pointer.

21. The digital data processor of claim 20, wherein said control logic alternately enables said second and third selectors, based on logic states of the least significant bit of the content of said main stack pointer, during the one-word push and pop operations.

22. The digital data processor of claim 21, wherein said second selector is enabled when the least significant bit of the content of said main stack pointer is logic >1'.

23. The digital data processor of claim 21, wherein said third selector is enabled when the least significant bit of the content of said main stack pointer is logic >0'.

24. The digital data processor of claim 17, wherein said control logic enables both said second and third selectors, irrespective of logic states of the least significant bit of the content of said main stack pointer, during the two-word push and pop operations.

25. The digital data processor of claim 20, wherein said control logic toggles the least significant bit of the content of said main stack pointer during the one-word push and pop operations.

Evidence Appendix

There is no evidence submitted pursuant to 37 CFR §§ 1.130, 1.131 or 1.132 or any other evidence entered by the examiner and relied upon by appellant in this Appeal.

Related Proceedings Appendix

None.